

# Metodologija razvoja IS

---

**Pilepić, Matija**

**Undergraduate thesis / Završni rad**

**2019**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **University of Rijeka / Sveučilište u Rijeci**

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:195:250611>

*Rights / Prava:* [In copyright](#)/[Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2024-07-12**



*Repository / Repozitorij:*

[Repository of the University of Rijeka, Faculty of Informatics and Digital Technologies - INFORI Repository](#)



Sveučilište u Rijeci – Odjel za informatiku

Preddiplomski jednopredmetni studij informatike

Matija Pilepić

# Metodologija razvoja IS

Završni rad

Mentor: prof. dr. sc. Mile Pavlić

Rijeka, 25.8.2019.

# SADRŽAJ

1. SAŽETAK.....	3
2. UVOD .....	4
3. OSNOVE O INFORMACIJSKIM SUSTAVIMA .....	5
3.1. OSNOVNI POJMOVI.....	5
3.2. INFORMACIJSKI SUSTAV .....	5
3.2 METODOLOGIJE ZA RAZVOJ IS .....	6
4. SPECIJALIZIRANA METODOLOGIJA MIRIS .....	7
4.1. MIRIS .....	7
4.2. MODEL PROCESA.....	10
4.2.1. METODA DEKOMPOZICIJE.....	10
4.2.2. DIJAGRAM TOKA PODATAKA.....	12
4.2.3. METODA ENTITETI-VEZE .....	16
4.3. PRAKTIČNI PRIMJER .....	22
5. ZAKLJUČAK.....	32
6. POPIS SLIKA .....	33
7. LITERATURA .....	34

# 1. SAŽETAK

U ovom radu pisati ću općenito o informacijskim sustavima, izložiti neke od najbitnijih definicija te Vas upoznati sa osnovnim pojmovima. Predstaviti ću Vam neke od specijaliziranih metodologija za razvoj informacijskih sustava, te Vas pobliže upoznati sa metodologijom MIRIS. Naglasak će biti na metodama modeliranja procesa i metodi entitete-veze, za koje ću predstaviti i praktičan primjer.

## 2. UVOD

Informacijski sustav tvrtke je prevažan za menadžment da bi se upravljanje njegovim razvojem prepustilo slučaju ili samo sektoru informatike. Za pokretanje stvaranja informacijsko sustava odgovoran je čelni čovjek tvrtke, a za njegovu realizaciju odgovoran je CIO<sup>1</sup>.

Kroz ovaj rad prikazat će se što je poslovni sustav, što je njegov informacijski sustav i dati pregled metoda za razvoj IS<sup>2</sup>.

Za razvoj IS se razvilo niz metoda koje su namijenjene što boljem, kvalitetnijem, ali i bržem razvoju tih sustava. U nastavku ću navesti neke od tih metoda, te se fokusirati na metodologiju MIRIS.

---

<sup>1</sup> Chief Information Officer

<sup>2</sup> Informacijski sustav

## 3. OSNOVE O INFORMACIJSKIM SUSTAVIMA

### 3.1. OSNOVNI POJMOVI

Sustav je skup elemenata, veza između elemenata te obilježja svrsishodno organiziranih za neki proces ili funkciju. Sustav je ukupnost načela ili stvari usklađenih i povezanih u cjelinu. Sustavi su organizacije kao NATO, Coca-Cola, sportski klub; ili organizmi ili dijelovi organizma ili nešto što za nas ima smisla promatrati kao cjelinu.

Za modeliranje IS je važno promatranje organizacijskih sustava, odnosno poslovnih organizacija.

Organizacija je svjesno udruživanje ljudi radi postizanja nekog cilja uz najmanji mogući napor. Organizacija rada svjesna je čovjekova djelatnost kojom se usklađuju svi činioци proizvodnje radi postizanja optimalnih rezultata rada. Poslovna organizacija je organizacija koja se bavi jednom ili s više poslovnih djelatnosti kao svojom misijom. Svaka poslovna organizacija ima informacijski sustav.

Poslovna djelatnost je sve ono što poslovna organizacija čini kako bi ostvarila prihode, bilo izravno ili posredno.

### 3.2. INFORMACIJSKI SUSTAV

Informacijski sustav se može definirati na nekoliko načina, navesti ću neke od definicija.

- Informacijski sustav je skup organiziranih i povezanih informacija koje tvore sustave i dio su toga istog sustava.
- Informacijski sustav je skup povezanih dijelova (softver, hardver, ljudi, procedure, informacije, te komunikacijske mreže) kojima je cilj pribaviti i prenijeti informacije i podatke za funkcioniranje, planiranje, odlučivanje i/ili upravljanje poslovnom organizacijom.
- Informacijski sustav jest sustav koji prikuplja, pohranjuje, čuva, obrađuje i isporučuje informacije važne za organizaciju, tako da budu dostupne i upotrebljive svakomu komu su potrebne. Informacijski sustav može se, ali ne mora, koristiti informacijskom tehnologijom (Čerić, 2004.).

Informacijski sustav jest sustav koji čine ljudi, programska i računalna oprema koja je napravljena, oblikovana i dovedena u operativno stanje te služi skupljanju, zapisivanju, spremanju i pronalaženju te prikazivanju informacija u odgovarajućem obliku (Kiš, 2002.).

Kao što je vidljivo, pojam informacijskog sustava je složen te ga nije moguće jasno i detaljno izreći sa jednom definicijom. Sa stajališta funkcije u organizacijskom sustavu, ima zadatak

trajno opskrbljivati potrebnim podacima poslovne procese i se razine upravljanja tim procesima te odlučivanja u tim procesima.

Temeljne funkcije IS su:

- prikupljanje i upis podataka u bazu podatak
- obrada podataka
- prikaz i ispostavljanje podataka iz baze podataka
- čuvanje podataka

### 3.2 METODOLOGIJE ZA RAZVOJ IS

Informacijski sustav je dio svake organizacije. Razvoj IS je zadaća informatičara raznih struka, a za sam razvoj postoje specijalizirane metodologije za razvoj IS. Navesti ću neke od tih metodologija.

Specijalizirane metodologije za razvoj IS:

- SSADM metodologija
- Specijalizirana metodologija MIRIS
- Metodologija IDEA
- Metodologija ORACLE\*CASE
- Metodologija IDEF
- Systems Development Life Cycle
- Business Systems Planning
- Information Engineering Methodology
- Agile Software Development
- Extreme Programming

Metodologija je puno, a ovo je samo mali dio njih. Informacijski sustav se može napraviti na različite načine, a od njih ne postoji najbolji način.

## 4. SPECIJALIZIRANA METODOLOGIJA MIRIS

### 4.1. MIRIS

Metodologija MIRIS (**M**etodologija za **R**azvoj **I**nformacijskih **S**ustava) je skup metoda čiji je cilj projektirati i izgraditi Informacijski sustav (IS). Metodologiju MIRIS razvio je prof. dr. sc. Mile Pavlič, a objavljena je 1995. godine (Pavlič, Informacijski sustavi, 2009).

Osnovna hipoteza oko koje je MIRIS oblikovan jest „dekompozicija sustava“, što znači da životni ciklus projektiranja dijelimo i tri faze.

1. Faza: apstraktno modeliranje sustava i podjela sustava u podsustave
2. Faza: modeliranje podsustava i propisivanje odgovarajuće metode za modeliranje procesa
3. Faza: modeliranje podataka procesa i definiranje arhitekture aplikacije prema modelu procesa i modelu podataka.

MIRIS se koristi sa tri osnovne metode: metodom za modeliranje podataka, metodom za modeliranje procesa i metodom za modeliranje aplikacija. Metode su slične mnogim metodama u drugim metodologijama.

Faze životnog ciklusa grupirane su u dvije skupine:

- logičko oblikovanje (projektiranje IS)
- fizičko oblikovanje (izgradnja IS)

Svaka skupina faza ima tri faze. Faze se dalje dijele u aktivnosti.

1. Strateško planiranje informacijskog sustava (SP)
  - 1.1. Analiza: Definiranje i obuka tima, dekompozicija procesa, popis dokumentacije i kretanje kroz sustav
  - 1.2. Podsustavi: Određivanje podsustava i veza
  - 1.3. Prioriteti: Određivanje prioriteta
  - 1.4. Resursi: Definiranje cjelovite infrastrukture
  - 1.5. Plan: Planiranje glavnih projekata i aktivnosti
2. Glavni projekt (GP)
  - 2.1. PZ: Izrada projektnog zadatka
  - 2.2. DTP: Intervjuiranje, raščlanjivanje i modeliranje procesa (DTP)
  - 2.3. Procesi GP: Analiza procesa, problema i prijedloga poboljšanja
  - 2.4. Podaci GP: Opisivanje podataka
  - 2.5. Plan GP: Planiranje izvedbenih projekata
  - 2.6. Resursi GP: Definiranje modela resursa glavnog projekta
3. Izvedbeni projekt (IP)
  - 3.1. DEV: Intervjuiranje, apstrakcija i modeliranje podataka (EV)
  - 3.2. Prevođenje: Prevođenje modela podataka u shemu BP (RM)
  - 3.3. Arhitektura IP: Definiranje arhitekture programskog proizvoda (APP)



- 3.4. Operacije IP: Projektiranje operacija nad shemom BP
- 4. Proizvodnja softvera (PS)
  - 4.1. Planiranje proizvodnje
    - 4.1.1. Planiranje aktivnosti proizvodnje SW
    - 4.1.2. Određivanje izvršitelja za pojedine zadatke i određivanje rokova
    - 4.1.3. Određivanje i kreiranje produkcijske, testne i razvojne okoline
  - 4.2. Oblikovanje baze podataka
    - 4.2.1. Prevođenje logičkog modela podataka u fizički model sheme baze podataka
    - 4.2.2. Kreiranje razvojne okoline za svakog pojedinog programera
    - 4.2.3. Punjenje sheme baze podataka u razvojnoj okolini iz postojeće BP
    - 4.2.4. Dodavanje novih koncepata iz modela podataka u razvojnu shemu BP (tip entiteta i dr.)
    - 4.2.5. Kreacija razvojne baze podataka
    - 4.2.6. Inicijalno punjenje testne baze podataka
  - 4.3. Razvoj programskog proizvoda
    - 4.3.1. Izgradnja glavnog izbornika (aplikacijskog stabla) ili dorada već postojećeg
    - 4.3.2. Izrada ekrana za pregled redaka svake tablice po jednom ili više ključeva
    - 4.3.3. Izrada ekrana za operacije nad jednim retkom tablice (unos, izmjena, brisanje i pregled)
    - 4.3.4. Izrada programskih modula različitih vrsta i namjena: obračuna, procedura, funkcija kontrola, look-upova nad tablicama (s prvim testiranjem modula)
    - 4.3.5. Izrada izvještaja (s prvim testiranjem modula)
  - 4.4. Testiranje u testnoj okolini
    - 4.4.1. Prijenos razvijenih programskih modula u testno okruženje
    - 4.4.2. Spajanje novih modula s postojećim
    - 4.4.3. Back-up verzija softvera
    - 4.4.4. Testiranje prototipa softvera nad testnom bazom podataka
    - 4.4.5. Ažuriranje planova proizvodnje softvera
    - 4.4.6. Izvođenje prema potrebi aktivnosti iz ranijih skupina aktivnosti i ponovno testiranje
  - 4.5. Testiranje i ispravljanje u radnoj okolini
    - 4.5.1. Prijenos razvijenih programskih modula u radno okruženje
    - 4.5.2. Spajanje novih modula s postojećima
    - 4.5.3. Back-up verzija softvera
    - 4.5.4. Punjenje baze podataka
    - 4.5.5. Testiranje prototipa softvera nad produkcijskom bazom podataka od strane programera
    - 4.5.6. Ažuriranje planova proizvodnje softvera
    - 4.5.7. Izvođenje prema potrebi aktivnosti iz ranijih skupina aktivnosti i ponovno testiranje
  - 4.6. Testiranje od strane korisnika
    - 4.6.1. Prezentacija softvera korisniku
    - 4.6.2. Testiranje od strane korisnika
    - 4.6.3. Izrada popisa primjedbi korisnika

- 4.6.4. Ažuriranje planova proizvodnje softvera
- 4.6.5. Izvođenje prema potrebi aktivnosti iz ranijih skupina aktivnosti i ponovno testiranje
- 4.6.6. Izrada zapisnika o testiranju i prihvaćanju faze uvođenja
- 5. Uvođenje (UVO)
  - 5.1. Instalacija gotovog softvera na produkcijsko okruženje (kod korisnika)
  - 5.2. Izrada uputa
  - 5.3. Prezentacija gotovog softvera
  - 5.4. Obuka
  - 5.5. Završne konverzije
  - 5.6. Završno testiranje
  - 5.7. Početak primjene nove aplikacije
  - 5.8. Uspostava novog sustava i potpisivanje primopredajnog zapisnika
- 6. Primjena i održavanje (ODR)
  - 6.1. Podešavanje novog aplikacijskog sustava
  - 6.2. Izvješće o procjeni novog projekta
  - 6.3. Raspodjela odgovornosti korisnika i programera
  - 6.4. Korištenje aplikacijskog sustava
  - 6.5. Postavljanje zahtjeva za izmjenama od strane korisnika

Izgradnja IS započinje izradom strateškog plana. Nakon toga se u fazi glavnog projekta analizira poslovanje i kreira model podataka.

U fazi izvedbenog projekta radi se model podataka i definira logička arhitektura programskoga proizvoda.

Faza proizvodnje započinje planiranjem proizvodnje, uzimajući u obzir postojeći programski proizvod i bazu podataka. Cilj je reorganizirati postojeću bazu podataka ili stvoriti novu ako ništa ne postoji i kreirati potrebni programski proizvod.

Gotov programski proizvod uvodi se u sustav, a zatim se primjenjuje u svakodnevnom radu.

Faze metodologije MIRIS se mogu preklapati i tijekom razvoja moguće se vratiti na bilo koju aktivnost, te je izmijeniti. Na taj način se prikuplja znanje o sustavu i proširuje model sustava, te upotpunjuje vizija potrebna za razvoj informacijskog sustava (Pavlic, Jakupović, & Čandrlić, Modeliranje Procesu, 2014).

## 4.2. MODEL PROCESA

Model procesa sastoji se od više različitih dijagrama i dokumenata koji opisuju sadržaj modela procesa (Pavlic, Jakupović, & Čandrlić, Modeliranje Procesu, 2014).

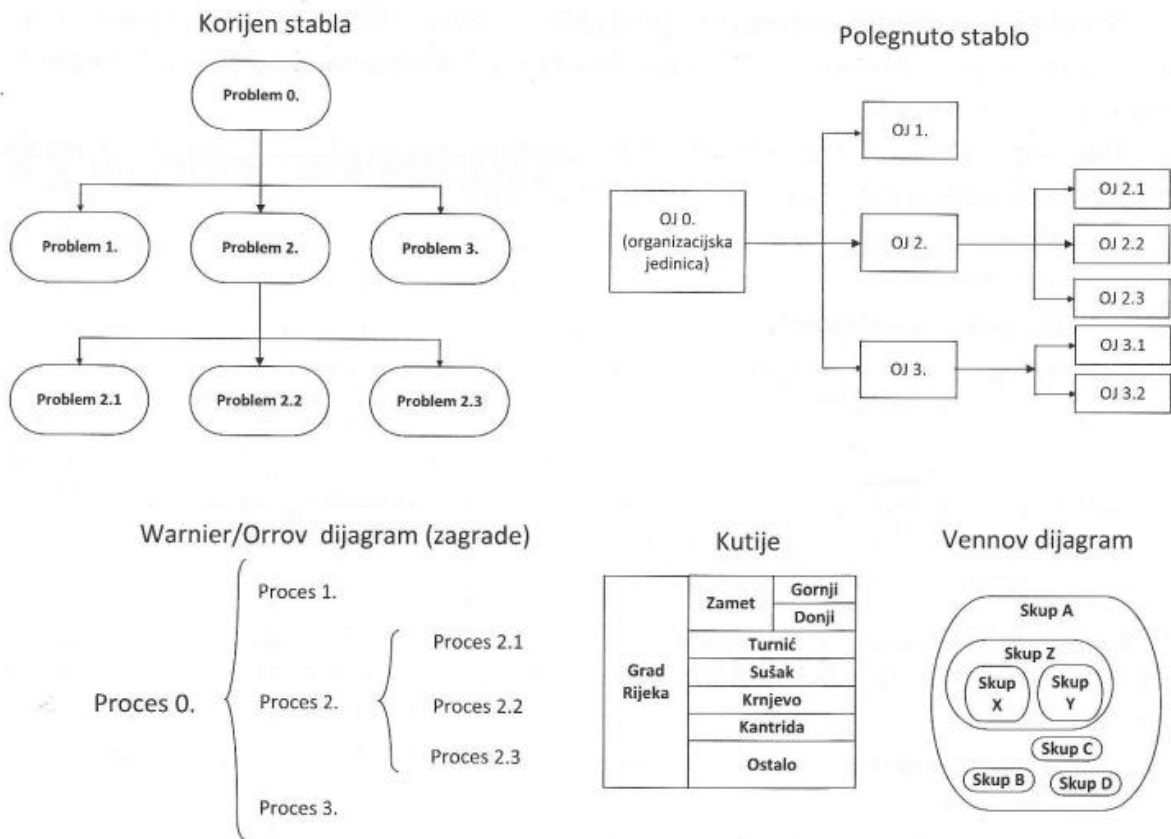
### 4.2.1. METODA DEKOMPOZICIJE

Dekompozicija je metoda rješavanja problema i opisivanja sustava počevši od konceptualno općih do detaljnih problema, odnosno od cjeline sustava k njegovim dijelovima (Pavlic, Jakupović, & Čandrlić, Modeliranje Procesu, 2014).

Dekompozicija je metoda savladavanja složenosti sustava kojom se problem rješava postupno od općega prema pojedinačnom (Pavlic, Jakupović, & Čandrlić, Modeliranje Procesu, 2014).

Dekompozicijom se složeni sustav rastavlja na više jednostavnijih podsustava koji se dalje mogu dekomponirati. Na taj način dolazimo do hijerarhijskog opisa sustava. Hijerarhijski možemo sustav opisati pomoću više modela pri čemu jedan model prikazuje cijeli sustav, dok ga drugi model još detaljnije opisuje. Postupkom dekompozicije dijelimo sustav na njegove podsustave, dok te podsustave možemo i dalje raščlanjivati i na taj način dolazimo do opisa sustava na raznim razinama detaljnosti, tj. dolazimo do hijerarhijskog opisa sustava (Pavlic, Jakupović, & Čandrlić, Modeliranje Procesu, 2014).

Dekompozicijski dijagrami, tj. dijagrami strukturalne raščlane daju prikaz cijele hijerarhije sustava. Takvi dijagrami opisuju sustav kao skup njegovih dijelova. Postoji više oblika dijagrama dekompozicije, međutim svi koriste istu tehniku raščlanjivanja. Dekompozicija je postupak raščlanjivanja sustava od općeg prema pojedinačnom, te onda od pojedinačnog prema općem (Pavlic, Jakupović, & Čandrlić, Modeliranje Procesu, 2014).



Slika 1 Prikaz dekompozicije

Na slici su prikazani dekompozicijski dijagrami različitih oblika. Iako su oni oblikom različiti, radi se o istoj tehnici koja jednako dobro može prikazati model dekompozicije.

U osnovi svih strukturnih tehnika je prikaz hijerarhijski više razine apstrakcije objekta promatranja pomoću niza strukturiranih objekata niže razine apstrakcije. Ustanovljen je niz grafički različitih načina (notacije, formi, konvencija) za prikaz. Pored ovih, postoje i druge notacije s nizom varijanti grafičkoga prikaza. Poželjno je da dekompozicijski dijagrami stanu na jednu stranicu A4 papira, iako to nije uvijek moguće (Pavlic, Jakupović, & Čandrić, Modeliranje Procesu, 2014).

#### 4.2.2. DIJAGRAM TOKA PODATAKA

Tekstualni zapis ima niz nedostataka, među kojima je najveća mana nemogućnost održavanja i pretraživanja informacija. Rješenje problema prikaza mnogobrojnih povezanih informacija leži u grafičkom prikazu informacija na više različitih razina apstrakcije, odnosno u metodi za modeliranje procesa.

Dijagram toka podataka (engl. Dataflow diagram, skraćeno DTP) je grafičko sredstvo za modeliranje i prezentaciju procesa sustava (Pavlic, Jakupović, & Čandrlić, Modeliranje Proces, 2014).

DTP je uveo De Marco 1978. godine kao proširenje tehnike „ulaz – proces – izlaz“ (engl. Input – Process – Output, skraćeno IPO) i tehnike HIPO (engl. hierarchial IPO) tvrtke IBM. Ideja metode HIPO je promatrati sustav i uočiti jedan ili nekoliko ulaza kojima se koristi neki proces, te jedan ili nekoliko izlaza iz tog procesa.

Odlike DTP-a:

- DTP je grafički prikaz procesa
- Korisnik i analitičar zajedno dolaze do modela
- Precizno se definiraju zahtjevi korisnika
- DTP je jezik za komunikaciju korisnika i analitičara
- Broj različitih vrsta grafičkih simbola na DTP-u je malen (četiri osnovna simbola).

Dijagram toka podataka sastoji se od (Pavlic, Jakupović, & Čandrlić, Modeliranje Proces, 2014):

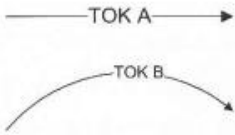
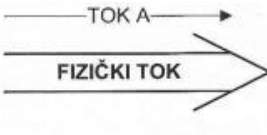
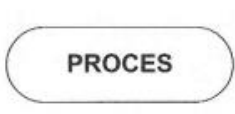
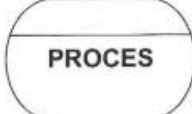
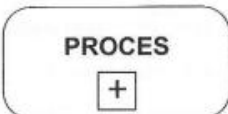
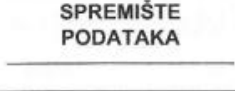
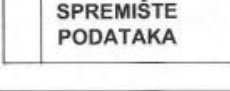
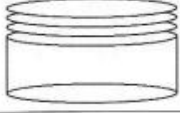
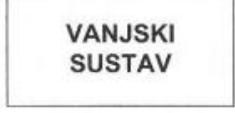
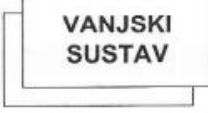


1. Ulaznih i izlaznih tokova podataka (izvještaja, dokumenata) koje sustav dobiva ili šalje u okolinu
2. vanjskih objekata (organizacija, drugih sustava) koji šalju podatke prema sustavu ili primaju tokove podataka od sustava
3. procesa sustava (programa, aktivnosti, podsustava) koji transformiraju ulazne u izlazne tokove podataka i
4. spremišta podataka (baze podataka) u kojima se čuvaju podaci potrebni za izvođenje procesa ili dobiveni kao rezultat rada procesa.

DTP je osnovna metoda analize za izradu modela procesa. DTP specificira što sustav radi. Osnovni cilj procesne analize (modeliranja procesa) je određivanje logičkog toka podataka kroz sustav. Procesi na dijagramu toka podataka se paralelno izvršavaju u stvarnom sustavu. DTP nije blok-dijagram (Flow-chart, dijagram toka programa) koji opisuje redoslijed izvršavanja nekog programa. Kod DTP-a se detaljno pomoću grafičkih simbola može opisati sustav, kako bi se mogao lakše, brže i u potpunosti shvatiti (Pavlic, Jakupović, & Čandrlić, Modeliranje Proces, 2014).

## Koncepti dijagrama toka podataka

DTP je grafičko sredstvo, a koristi se minimalnim brojem čovjeku bliskih koncepata (elemenata, grafičkih simbola) za opis sustava. Općenito, bilo koji grafički simbol od koga se gradi DTP nazivamo konceptom (Pavlic, Jakupović, & Čandrić, Modeliranje Procesu, 2014).

Postoje različite grafičke inačice prikaza koncepata dijagrama toka podataka. Iako su simboli grafički različiti, oni čine isti logički skup koncepata.

KONCEPT	SIMBOL po DeMarcu, Yourdonu	SIMBOL po Ganeu i Sarsonici	SIMBOL po BPMN-u
TOK PODATAKA predstavlja se vektorom ili usmjerenim lukom			Tijek sekvence Tijek poruke Pridruživanje Pridruživanje podataka
PROCES (funkcija) predstavlja se ovalom, elipsom, krugom i slično			
SPREMIŠTE (skladište) podataka predstavlja se s dvjema paralelnim crtama			
VANJSKI SUSTAV (izvorište ili odredište, granični entitet) predstavlja se pravokutnikom			
DOGAĐAJ predstavlja se krugom	nema	nema	

Slika 2 Grafički oblik koncepata strukture DTP-a

### Vanjski sustav

Vanjski sustav je koncept koji reprezentira sustave koji su u vezi s promatranim sustavom, bilo da su izvori ili odredišta podataka, a čiji nas procesi ne zanimaju.

Vanjski se sustav još naziva i vanjsko izvorište ili odredište podataka.

Izvorište ili odredište podataka su na primjer: osoba, poduzeće, organizacijska jedinica, ustanova, država, banka, sud...

## Tok podataka

Tok podataka je skup podataka koji se kreću od jednog dijela sustava k drugom te ih istovremeno povezuju i koji ima jasnu semantiku i strukturu.

„Tok podataka je cjevovod kroz koji teku informacije“.

Svaki tok podataka ima naziv na dijagramu. Imena se biraju tako da jasno prezentiraju podatke i ono što se o njima zna. Nazivi tokova su imenice, najčešće u jednini, ili kombinacije imenica i pridjeva. Nije dopušteno postojanje različitih tokova podataka s istim imenom. Svaki tok podataka ima jedinstveno ime.

Tokovi podataka prema spremištu podataka mogu imati isti naziv kao i naziv spremišta podataka te se ne trebaju imenovati. To je moguće samo u slučaju kad je sadržaj toka podataka jednak sadržaju spremišta podataka.

Tok podataka može biti dokument u sustavu (faktura, ček, narudžba), skup dokumenata (plan proizvodnje, projektna dokumentacija), elementarni podatak (tečaj dolara, dan primopredaje), bilo koje znanje (prijedlog o najboljem proizvođaču, neformaliziran zahtjev).

Tok podataka povezuje komponente dijagrama toka podataka, pa je tok podataka komponenta veze sustava, odnosno koncept za povezivanje drugih koncepata modela.

Tokovi podataka su mjerilo za formalnu organiziranost sustava. Može se zaključiti da je sustav dobro formalno organiziran ako su njegovi tokovi podataka dokumentirani.

## Spremište podataka

Spremište (skladište) podataka je koncept za prikaz grupa podataka. Spremište podataka je memorijsko mjesto gdje su podaci pohranjeni (gdje se čuvaju) za kasnije procese obrade.

Spremište podataka je jedan ili više tokova podataka u mirovanju, odloženi (akumulirani) sadržaj toka podataka za buduću uporabu procesa.

Spremište podataka može biti dokument (upit, ponuda, izvještaj), skup dokumenata (plan proizvodnje, projektna dokumentacija), datoteka ili baza podataka (bilo kakav skup podataka na računalu) ne formatizirani skup podataka (knjiga o projektiranju IS-a), znanje u ljudskom umu, biblioteka, skriveni skup podataka, bilo koje znanje.

Ime spremišta podataka se određuje isto kao i ime toka podataka. Spremište i tok podataka su koncepti sadržajno slične semantike s tim da tok ima još i ulogu povezivanja koncepata.

Svako spremište mora biti punjeno nekim procesom na nekom dijagramu u cjelokupnome modelu i nije moguće iz spremišta podataka uzeti neke podatke koji nisu pohranjeni u njega na nekome mjestu u modelu i u sustavu. Ako se modelira samo dio sustava, onda je moguće da se spremište puni u drugome dijelu.

Općenito, postoje dvije operacije koje proces može izvršiti nad spremištem podataka:

1. pisanje u spremište podataka
2. čitanje iz spremišta podataka

pri čemu pisanje prethodi čitanju.

Spremište podataka je riznica podataka, odnosno akumulacija podataka, odloženih sadržaja tokova podataka.

## Proces

Proces je skup povezanih aktivnosti i odluka preko kojih objekti sustava ostvaruju dijelove cilja svoga postojanja, a za njihovo izvršenje su potrebni određeni resursi i određeno vrijeme (Pavlic, Jakupović, & Čandrlić, Modeliranje Procesa, 2014).

Proces je transformacija strukture i sadržaja ulaznih u izlazne tokove podataka. Sinonim za proces su: rad, aktivnost, posao, operacija. Proces je neka količina posla izvršena s podacima. Proces na dijagramu toka podataka može biti u vezi sa i samo sa tokom podataka koji mu donosi podatke ili ih odnosi nakon transformacije. Svaki proces mora proizvoditi bar jedan izlazni tok podataka i imati bar jedan ulazni tok podataka, a može ih biti i više.

Svaki proces ima naziv procesa. Naziv je kratak opis značenja procesa, ne i precizna specifikacija njegove unutarnje logike. Za ime se bira glagol, glagolska imenica ili skup riječi koji opisuju vrstu posla.

Da bi proces mogao transformirati ulaz u izlaz, on ima neku unutarnju logiku transformacije. Ta logika je suština procesa. Logika procesa se detaljno opisuje nekim od sredstava za opis logike procesa. Metodike detaljno propisuju koji se procesi kako opisuju.



#### 4.2.3. METODA ENTITETI-VEZE

Metoda entiteti-veze (skraćeno EV) je grafički prikaz međusobno povezanih grupa podataka promatranoga sustava. EV je semantički bogata metoda za modeliranje podataka jer raspolaže ljudski bliskim konceptima. EV se odlikuje prirodnošću opisa a njezini koncepti su bliski korisniku, pa je shema modela podataka laka za razumijevanje i komunikaciju korisnika i projektanta (Pavlic, Oblikovanje baza podataka, 2011).

Razvojem EV-a nastalo je niz varijanti polazne metode i ona je postala jedna od najčešće korištenih metoda. Tako se EV koristi u specijaliziranim metodologijama CASE\*Method, MIRIS, SSADM, IEM i dr.

Za modeliranje podataka postoje i često korištene standardizirane notacije kao što su ISO UML (class diagrams) i IEEE IDEF1x.

Koncepti strukture metode entiteti-veze

Osnovni koncepti metode entiteti-veze od kojih se gradi struktura modela entiteti-veze su:

- Entitet i tip entiteta
- Veza i tip veze
- Atribut tipa entiteta
- Slab tip entiteta i specijalni tipovi veza
- Agregirani tip entiteta
- Povratni tip veze
- Generalizacijski tip veze

Model podataka metodom EV gradi se uporabom grafičkih simbola. Osnovni simboli za gradnju DEV-a dani su na slici:

KONCEPT	SIMBOL	PRIMJER
TIP ENTITETA		
SLAB TIP ENTITETA		
TIP VEZE		
ATRIBUT		
AGREGACIJA		
POVRATNA VEZA		
GENERALIZACIJA		

Slika 3 Grafički oblik konceptata strukture EV metode

DEV je grafički prikaz modela podataka sustava, metodom EV.

Podmodel je dio cjelokupnog modela podataka prikazan na jednoj od niza slika DEV-a, a sadrži odabrane tipove entiteta, tipove veza i attribute jedne kohezivne cjeline.

Entitet je pojam u poslovanju o kome se zahtjeva čuvanje nekih podataka.

Entitet je stvar koja ima stvarno ili pojedinačno postojanje u stvarnosti ili mislima.

Primjeri entiteta: grad Zagreb, osoba, pojedini kupac, itd. Slični entiteti koji imaju zajednička svojstva se klasificiraju u tipove entiteta.

Klasifikacija, razvrstavanje na temelju nekog sustava; podjela predmeta ili pojmova na razrede, skupine, rodove, vrste, tipove, prema određenim načelima koja ovise o vrsti građe i svrsi klasifikacije.

Tip entiteta je skup sličnih pojedinačnih entiteta dobiven procesom klasifikacije apstrakcije. To je skup entiteta istoga tipa.

Klasa entiteta je skup pojedinačnih pojavljivanja entiteta.

Tip entiteta je imenovana klasa entiteta.

Atribut tipa entiteta je funkcija koja preslikava tip entiteta u tip vrijednosti ili u kartezijev produkt tipa vrijednosti.

Atribut je imenovana karakteristika nekog entiteta. Ovo svojstvo je imenovana funkcija koja pridružuje tom entitetu neku vrijednost iz tipa vrijednosti. Tip vrijednosti povezan je s tipom entiteta preko atributa.

Primjeri atributa: boja, datum rođenja, ime, prezime i slično.

Ključ tipa entiteta je takav skup atributa koji neovisno o vremenu zadovoljava uvjet jedinstvenosti i uvjet neredundantnosti.

Uvjet jedinstvenosti:

Ne postoje dva pojedinačna pojavljivanja entiteta u tipu entiteta takva da imaju istu vrijednost atributa koji čine ključ i ne postoje dva tipa entiteta koji imaju isti skup atributa za ključ.

Uvjet neredundantnosti:

Ako se iz ključa izostavi bilo koji od atributa, uvjet neredundantnosti se gubi. Ne postoji niti jedan atribut kao dio ključa koji se može izostaviti, da se pritom uvjet jedinstvenosti ne gubi.

Primarni ključ je jedan od ključeva tipa entiteta koji je odabran zbog nekog najvažnijeg razloga za zamjenika entiteta toga tipa (za ključ). Jedan od razloga je njegovo često korištenje. U EV metodi uvodimo samo jedan ključ za svaki tip entiteta i to primarni (glavni) ključ. Ovaj će ključni atribut postati osnova za fizičko zapisivanje podataka u bazu podataka.

Veza (asocijacija, povezivanje) je pridruživanje između entiteta.

Veza predstavlja odnos koji postoji među entitetima bilo u stvarnosti, bilo u mislima. Entiteti se nalaze u različitim odnosima, a te odnose nazivamo veze.

Tip veze je skup veza između istih tipova entiteta.

Red tipa veze je broj tipova entiteta povezanih tim tipom veze.

Naziv tipa veze je riječ ili grupa riječi koji predstavljaju radnju (glagoli, glagolske imenice), operaciju ili odnos među entitetima, a koji nam jasno prikazuju u kom odnosu su tipovi entiteta.

Uloga entiteta u vezi je imenovana funkcija koja pridružuje taj entitet nekom drugom entitetu. U uređenome paru neke veze svaki entitet ima neku ulogu.

Tip veze sastoji se od dvaju preslikavanja (koja nazivamo uloge) između tipova entiteta. Svaki tip veze ime dvije uloge što znači da se sastoji od dva preslikavanja prvog tipa entiteta na drugi i obrnuto. Svako preslikavanje ograničeno je dopuštenim brojem povezanih entiteta. Svaka pojedina veza iz nekog tipa veze povezuje entitete dvaju tipova entiteta (Pavlic, Oblikovanje baza podataka, 2011).

Brojnost tipa veze je broj koji kaže koliko entiteta pojedinog tipa entiteta sudjeluje u tipu veze s entitetom iz drugog tipa entiteta (Pavlic, Oblikovanje baza podataka, 2011).

Moguće je odrediti donju (DG) i gornju granicu (GG) brojnosti za obje uloge, tj.:

- U koliko će se najmanje i najviše pojedinačnih pojavljivanja tipa entiteta E preslikati pojavljivanje tipa entiteta E1?

- U koliko će se najmanje i najviše pojedinačnih pojavljivanja tipa entiteta E1 preslikati pojavljivanje tipa entiteta E?

Oba broja, najmanji i najveći broj preslikavanja, upisujemo na dijagram, odvojene zarezom i upisane u zagrade kao uređeni par (DG,GG). Uređeni par (DG,GG) nazivamo brojnost preslikavanja uloge. Brojnost preslikavanja tipa veze su dva uređena para oblika (DG,GG) : (DG, GG), pri čemu prvi par pripada jednoj ulozi, a drugi drugoj ulozi (Pavlic, Oblikovanje baza podataka, 2011).

### Jak i slab tip entiteta

Jak tip entiteta je tip entiteta koji ima vlastiti primarni ključ i nije ovisan o drugim tipovima entiteta u modelu podataka.

Slab tip entiteta je tip entiteta koji je na neki način ovisan o nekom drugome tipu entiteta, a ta se ovisnost prikazuje specijalnim tipom veze među jakim i slabim tipom entiteta.

Specijalni tip veze je tip veze koji povezuje dva tipa entiteta među čijim entitetima postoji ovisnost. Moguće su različite vrste ovisnosti, a prema njoj definira se vrsta specijalnog tipa veze.

Različite vrste specijalnih tipova veze su:

- Egzistencijalna (E)
- Identifikacijska (I)
- Egzistencijalna i identifikacijska (E&I)
- Prethođenja (P)
- Generalizacijska (G)
- Hijerarhijska (H)
- Agregacijska (sastavljena od dvije E&I između triju tipova entiteta)

Egzistencijalni tip veze je tip veze koji povezuje dva tipa entiteta od kojih jedan egzistencijalno ovisi o drugom. Egzistencijalno ovisan tip entiteta je slab tip entiteta u odnosu na tip entiteta o kome ovisi, koji zovemo jak tip entiteta.

Identifikacijski tip veze je tip veze koji označava da se jedan tip entiteta u modelu ne može jedinstveno identificirati vlastitim ključnim atributima, već se mora koristiti identifikator nekog drugog tipa entiteta da bi se postigla jedinstvena identifikacija.

Egzistencijalni i identifikacijski tip veze, dva su tipa entiteta povezana specijalnim tipom veze E&I ako je jedan tip entiteta istovremeno i egzistencijalno i identifikacijski ovisan o drugome tipu entiteta.

Tip veze prethođenja. Tip specijalne veze je tip veze prethođenja među tipovima entiteta ako postojanju jednog tipa entiteta (ne obavezno slabog) prethodi postojanje drugog tipa entiteta.

Neki tipovi entiteta su podtipovi nekog drugog tipa entiteta, koga zovemo nadtip. Nadtip i podtip povezani su generalizacijskim tipom veze.

Povratni tip veze (rekurzivni tip veze) je povratni tip veze ako povezuje jedan tip entiteta sam sa sobom.

Agregacija je apstrakcija u kojoj se tip veze između dvaju ili više tipova entiteta tretira kao novi tip entiteta.

Tip veze postaje tip agregirani tip entiteta ako (Pavlic, Oblikovanje baza podataka, 2011):

- Pojavljuje se atribut tipa veze, tj. atribut ne može pripadati niti jednom pojedinačnom tipu entiteta
- Potrebno je međusobno povezati dva tipa entiteta, npr. veza Student polaze ispit iz kolegija treba biti povezana s vezom Ispituje Profesor
- Potrebno je razdvojiti višestruki u binarni tip veze, npr. veza Isporučka između entiteta Roba, Dobavljač, Kupac, Proizvođač, Dostavljač i Skladište.

- Brojnost tipa veze po gornjoj granici s obje strane iznosi  $M$  (na primjer, jedan radnik može poznavati više stranih jezika ( $M$ ), dok jedan strani jezik može poznavati više radnika ( $M$ ))

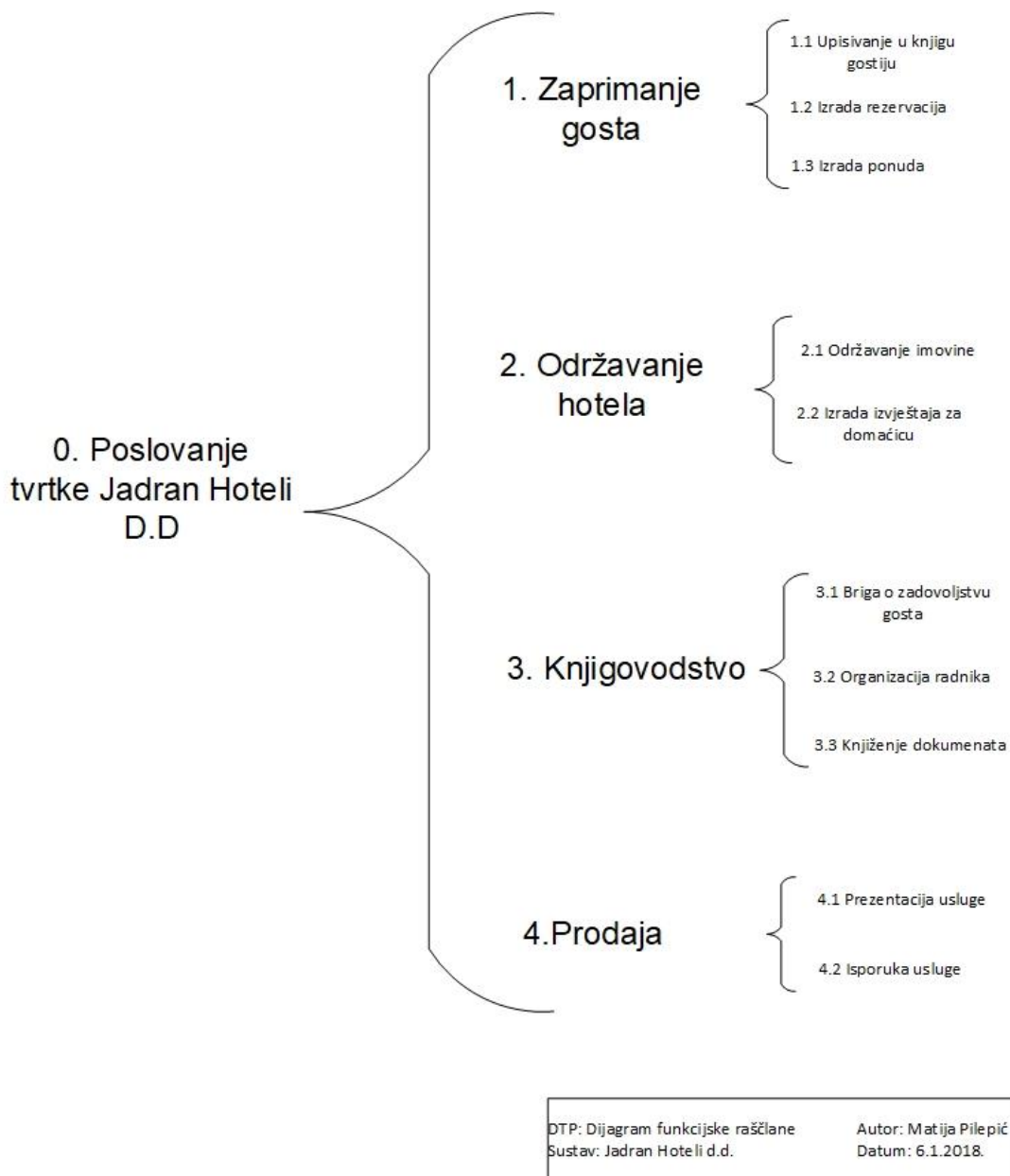
Agregirani tip entiteta nastaje od tipa veze u trenutku kada se pojavi atribut toga tipa veze. Svaki tip veze koji ima bar jedan atribut također postaje agregacija, odnosno agregirani tip entiteta. Agregirani tip entiteta je tip veze koji je postao tip entiteta. Tip veze koji ima atribut je tip entiteta, a ne tip veze zato što agregacija pri prevođenju u relacijsku shemu postaje relacija koja sadrži podatke potrebne za paćenje poslovanja. Tip veze ne postaje relacija.

Ključ agregacije je skup ključnih atributa komponenti agregacije. Agregacija dobiva ključ od komponenti agregacije. Brojnosti ključa tipa agregacije uvijek je  $(1,1) : (1:1)$  inače to ne bi bio ključ.

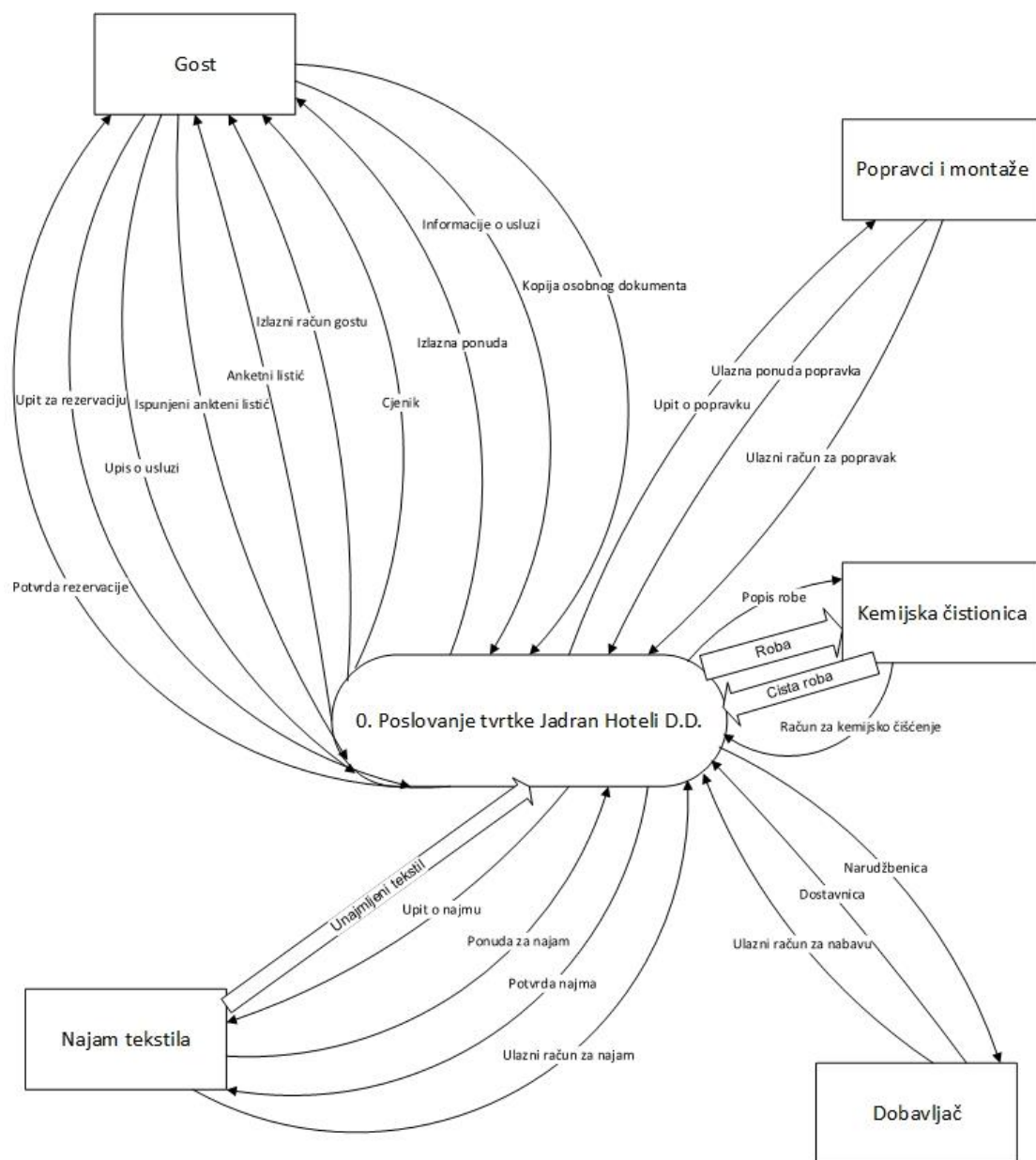
### 4.3. PRAKTIČNI PRIMJER

Tvrtka Jadran hoteli d.d. sa sjedištem u Rijeci, kao osnovnom djelatnosti, bavi se pružanjem usluga smještaja i ostalih ugostiteljskih usluga karakterističnih za turističku (ili hotelijersku) djelatnost i to na području Rijeke, Kostrene i Kraljevice.

Na sljedećim slikama biti će prikazan dijagram funkcijske raščlambe, dijagram konteksta, dijagrami toka podataka od razine 0 do razine 4.



Slika 4 Dijagram funkcijske raščlambe

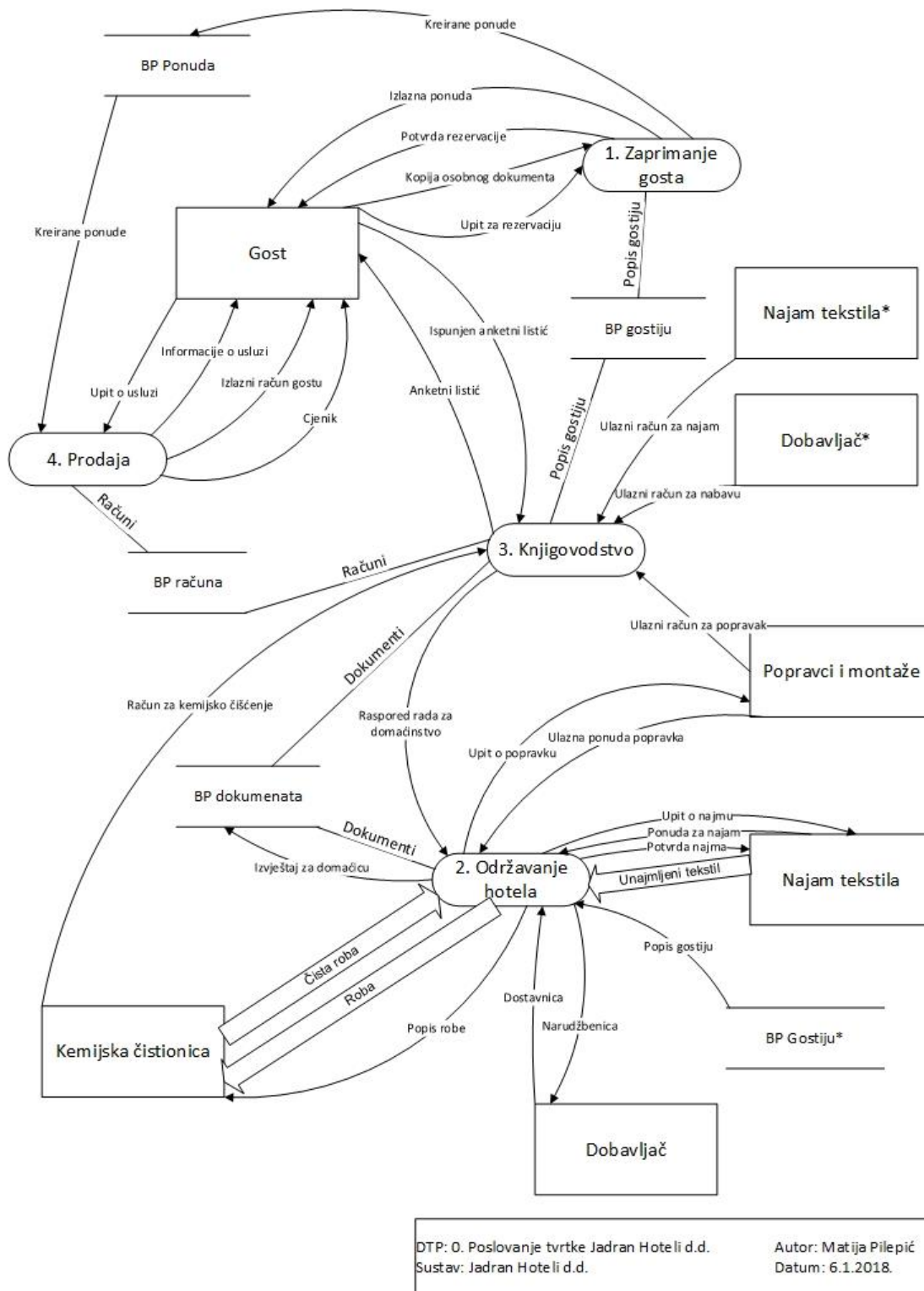


DTP: Dijagram Konteksta  
Sustav: Jadran Hoteli d.d.

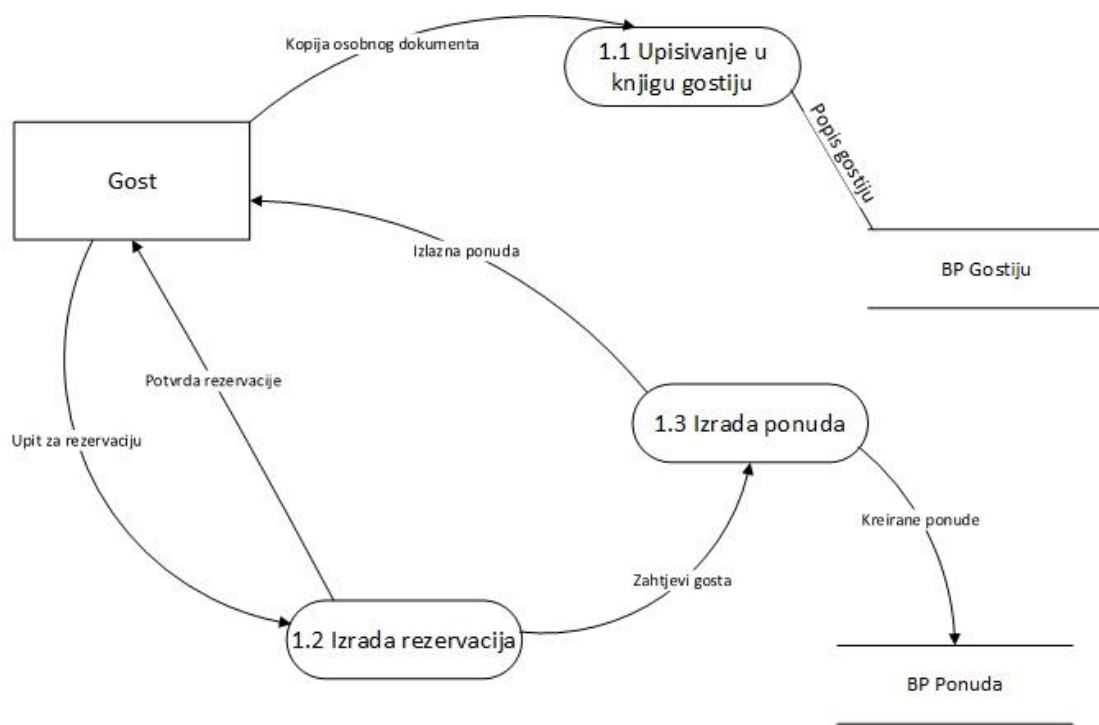
Autor: Matija Pilepić  
Datum: 6.1.2018.

Slika 5 Dijagram konteksta



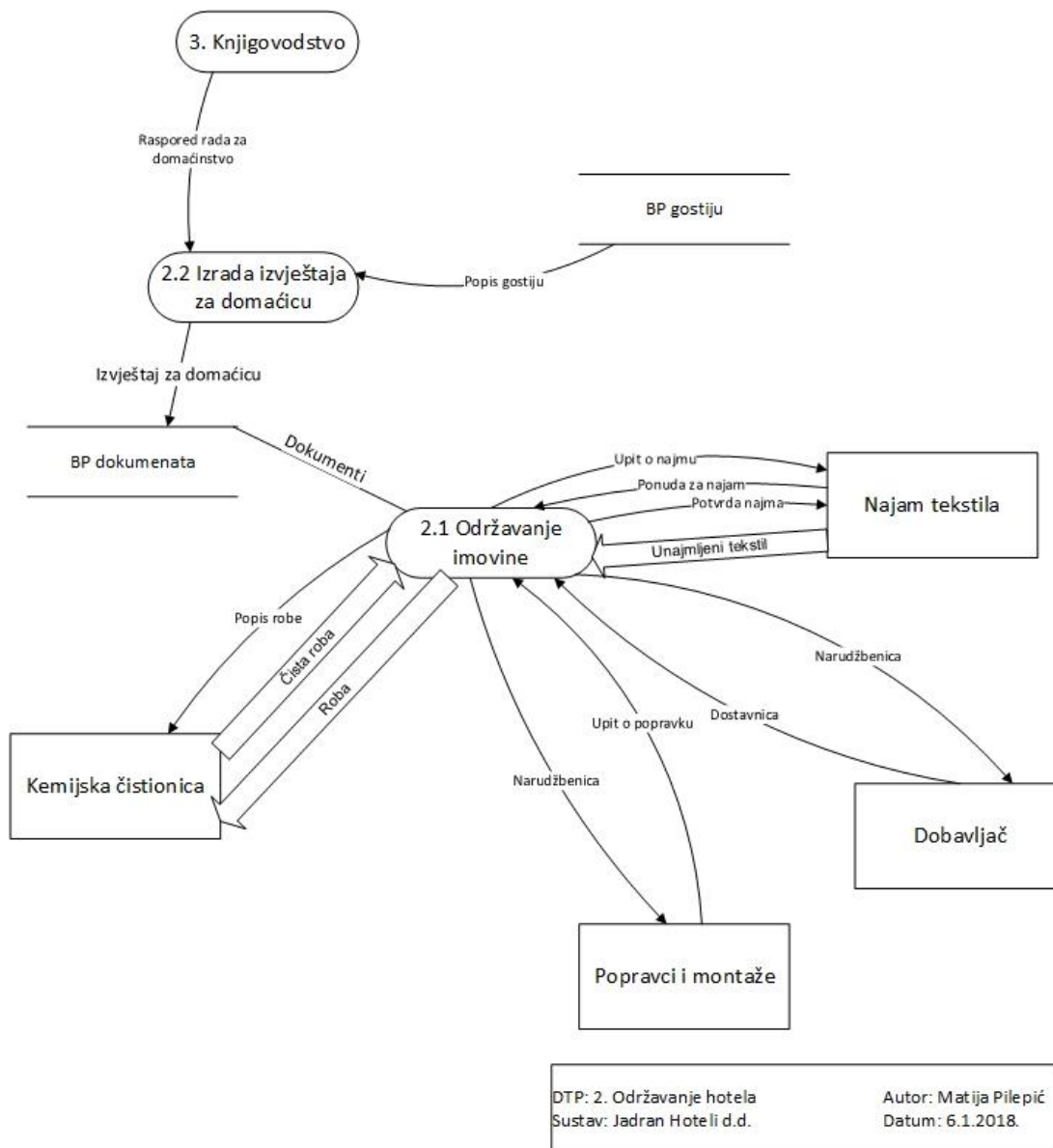


Slika 6 DTP 0.

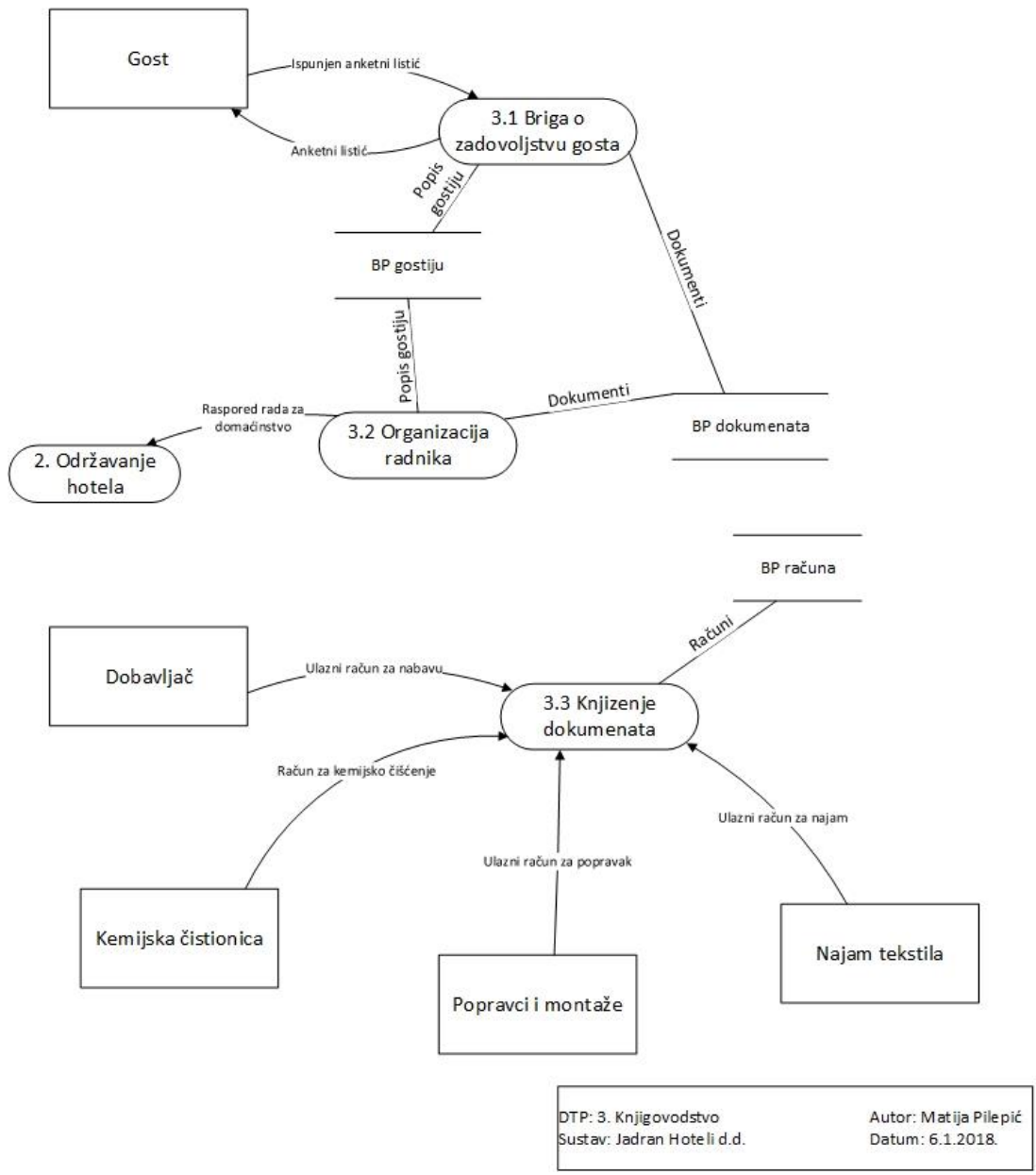


DTP: 1. Zaprimanje gosta  
 Sustav: Jadran Hoteli d.d.  
 Autor: Matija Pilepić  
 Datum: 6.1.2018.

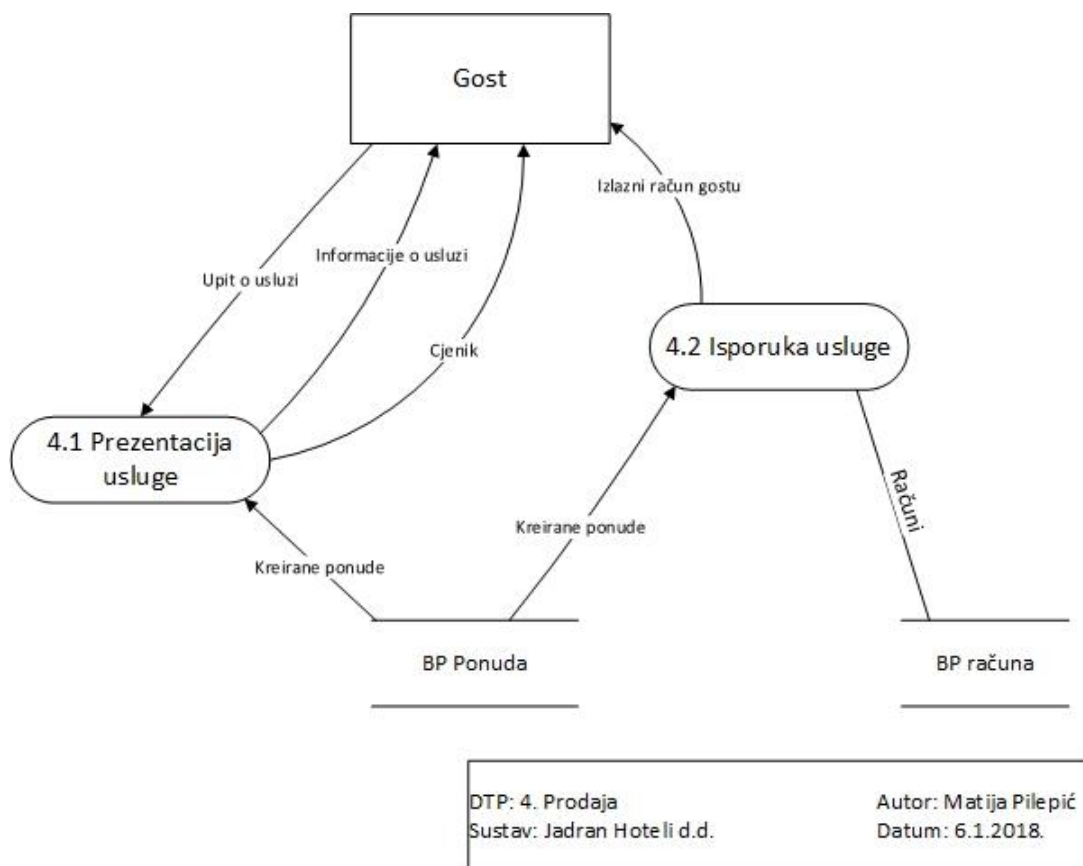
Slika 7 DTP 1.



Slika 8 DTP 2.



Slika 9 DTP 3.



Slika 10 DTP 4.

Stranica: \_\_\_\_\_

**JURJEVIĆ D.O.O. STARCI 5**

51000 RIJEKA MB:02438330  
0917941452 M, STIPE 0917939340  
Telefoni : 639 614, 091/7941-452

Član uprave: MARINA JURJEVIĆ  
Sud : TRGOVAČKI SUD U RIJECI  
Mbs : 04025257 Temejni kapital : 20.000,00

OIB: 94358262618

**PONUĐA 32/2016**

RIJEKA,20.09.2016

Datum : 20.09.2016  
Vrijednost ponude do : 20.09.2016  
Mjesto isporuke :  
Rok isporuke :

JADRAN HOTELI "HOTEL JADRAN"  
STROSSMAYEROVA 1  
51000 RIJEKA  
HRVATSKA

Način plaćanja :  
Poziv na broj plaćanja 05 19-2016-32  
Žiro račun (IBAN) : HR3223400091110354385  
Fax :

red. br.	šifra artikla	Naziv artikla-usluge	j.mj.	količina	cijena	vrijednost	Porez %
1.	90024	USLUGA IZRADA ZAŠTITNE TENDE U RECEPCIJI I RESTORANU KOMADA 9	M2	78,000	200,00	15.600,00	25,00

Ukupna vrijednost : 15.600,00

Ukupno bez poreza : 15.600,00 KN  
Ukupno PDV : 3.900,00 KN  
UKUPNO SA PDV-om : 19.500,00 KN

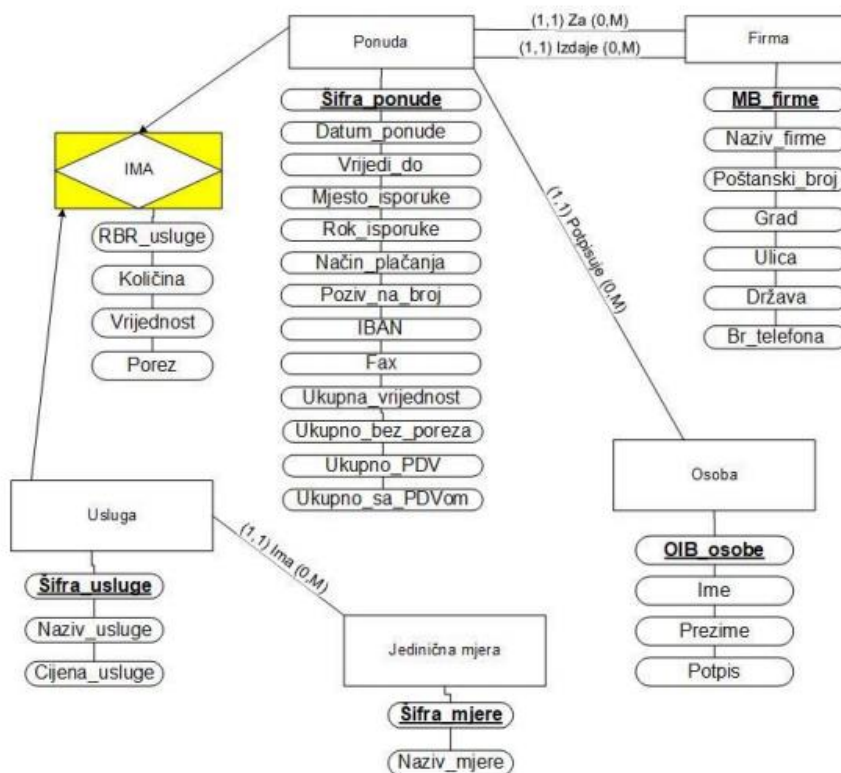
U CIJENU JE URAČUNATA MONTAŽA I DEMONTAŽA

Izradio :

Odgovorna osoba : MARINA JURJEVIĆ

**JURJEVIĆ**  
d.o.o. RIJEKA  
OIB: 94358262618

Slika 11 Ulazna ponuda



DEV ulazna ponuda  
 Jadran Hoteli d.d.  
 5.5.2018.  
 Matija Pilepić

Slika 12 DEV Ulazna ponuda

Relacijski model

**Usluga**(RBR\_usluge, Šifra\_usluge, Naziv\_usluge, Cijena\_usluge, Šifra\_mjere)

**Ima**(Šifra\_usluge, Šifra\_ponude, RBR\_usluge, Količina, Vrijednost, Porez)

**Ponuda**(Šifra\_ponude, Datum\_ponude, Vrijedi\_do, Mjesto\_isporuke, Rok\_isporuke, Način\_plaćanja, Poziv\_na\_broj, IBAN, Fax, Ukupna\_vrijednost, Ukupno\_bez\_poreza, Ukupno\_PDV, Ukupno\_sa\_PDVom, MB\_Firme, OIB\_osobe)

**Firma** (MB\_Firme, OIB\_Firme, Naziv\_firme, Grad, Ulica, Država, Br\_telefona, Porezni\_broj, Opis, Email)

**Jedinična mjera** (Šifra\_mjere, Naziv\_mjere)

**Osoba**(OIB\_Osobe, Ime, Prezime, Potpis)



## 5. ZAKLJUČAK

Nakon ovoga rada, možemo zaključiti koliko je informacijski sustav bitan u poslovanju neke tvrtke, te se njegova važnost ne smije zanemarivati. Metodologija za razvoj informacijskog sustava ima mnogo, te je teško, možda i nemoguće reći koja od njih je najbolja. Metodologija MIRIS sigurno je jedno od poznatijih metodologija i često je upotrebljavana u praksi. MIRIS se sastoji od 6 faza koje se međusobno mogu isprepletati, te je moguće vraćati se sa faze na neku prethodnu, što mislim da je jako bitno jer na taj način prikupljamo znanja o sustavu i možemo još dodatno proširiti model sustava i upotpuniti svoju viziju potrebnu za razvoj informacijskog sustava. Na kraju rada priložio sam i praktični primjer u kojemu su modelirani dijagram funkcijske raščlambe, dijagram konteksta, i dijagrami toka podataka 4 razine te jedan dokument obrađen metodom entiteti-veze.

## 6. POPIS SLIKA

Slika 1 Prikaz dekompozicije .....	11
Slika 2 Grafički oblik koncepata strukture DTP-a .....	13
Slika 3 Grafički oblik koncepata strukture EV metode.....	17
Slika 4 Dijagram funkcijske raščlambe.....	22
Slika 5 Dijagram konteksta .....	23
Slika 6 DTP 0. ....	24
Slika 7 DTP 1. ....	25
Slika 8 DTP 2. ....	26
Slika 9 DTP 3. ....	27
Slika 10 DTP 4. ....	28
Slika 11 Ulazna ponuda.....	29
Slika 12 DEV Ulazna ponuda .....	30

## 7. LITERATURA

1. Pavlić, M., (1996): Razvoj informacijskih sustava – projektiranje, praktična iskustva, metodologija, Znak, Zagreb.
2. Pavlić, M., "Informacijski sustavi", Školska knjiga, Zagreb, 2011.
3. Pavlić, M., "Oblikovanje baza podataka", Odjel za informatiku Sveučilišta u Rijeci, Rijeka, 2011.
4. Pavlić, Mile; Jakupović, Alen; Čandrlić, Sanja, Modeliranje procesa / Martina Ašenbrener Katić (ur.), Rijeka: Odjel za informatiku Sveučilišta u Rijeci, 2014.
5. Mile Pavlić, blog, <http://milepavlic.blogspot.com>
6. Wikipedia, [https://hr.wikipedia.org/wiki/Informacijski\\_sustavi](https://hr.wikipedia.org/wiki/Informacijski_sustavi)